



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian sebelumnya

Penelitian – penelitian terdahulu berfungsi sebagai pendukung untuk melakukan penelitian. Penelitian – penelitian sebelumnya telah mengkaji masalah pembagian kerja dan upah yang masing – masing berpengaruh terhadap prestasi kerja, dan beberapa penelitian lain yang masih memiliki kaitan dengan variabel dalam penelitian ini.

- a. Penelitian pertama dilakukan oleh Dhebys Suryani, Eka Larasati Amalia yang berjudul “Aplikasi *Chatbot* objek wisata Jawa Timur berbasis AIML” yang di arsip oleh SMARTICS Journal Vol.3, No.2, Oktober 2017 di Politeknik Negeri Malang program studi teknik informatika.

Hasil penelitian: Penelitian sebelumnya terkait dengan *chatbot* yang diteliti pada jurnal ini dikhususkan mengenai aplikasi *chatbot* objek wisata Jawa Timur berbasis AIML, pada penelitian ini dilakukan analisa kebutuhan sistem, analisa pengguna sistem, perancangan sistem, dan yang terakhir barulah dilakukan implementasi dan pengujian untuk mengetahui kecocokan *input* yang dimasukkan pengguna dengan jawaban yang diberikan oleh sistem.

- b. Penelitian yang kedua dilakukan oleh Elisabet Nila S.C.P, Irawan Afrianto dengan judul “ Rancang Bangun Aplikasi *Chatbot* Informasi

Objek Wisata Kota Bandung Dengan Pendekatan *Natural Language Processing*". Bandung, 2015 di Universitas Komputer Indonesia Jurnal Ilmiah Komputer dan Informatika (KOMPUTA).

Hasil penelitian: Pada penelitian ini dilakukan pembangunan sebuah aplikasi *chatbot* untuk memberikan informasi objek wisata kota Bandung bagi para wisatawan maupun penduduk lokal. Aplikasi ini menggunakan *Natural Language Processing* atau bahasa alami dengan metode *Text Mining*, serta Pemodelan sistem *Unified Model Language*. Tujuan dari pembuatan aplikasi ini adalah memberikan informasi bagi pengunjung mengenai tempat-tempat wisata yang ada di Bandung.

Dapat disimpulkan bahwa pengunjung setuju aplikasi *chatbot* ini dapat memudahkan pengunjung untuk mencari informasi mengenai alamat tempat-tempat wisata yang ada di kota Bandung. Tanya jawab atau percakapan yang terjadi seakan-akan dilakukan oleh dua pribadi manusia yang saling berkomunikasi. Sesuatu yang dihasilkan dari penelitian ini adalah : Pendekatan *Natural Language Processing* atau NLP pada *chatbot* yang terjadi layaknya dilakukan oleh manusia dengan manusia, penggunaan metode *text mining* juga memudahkan dalam pencarian *keyword*, diketahui bahwa percakapan ini dilakukan menggunakan internet sehingga dapat dilakukan dimana pun dan kapan saja, yang terakhir aplikasi *chatbot* ini memberikan informasi dalam hal pencarian informasi yang jelas mengenai objek wisata yang ada di Bandung.

Persamaan dari kedua jurnal ini adalah perancangan sistem *chatbot*. Perbedaan pada penelitian sebelumnya dengan penelitian ini ada pada *platform* nya penelitian ini menggunakan *platform line*, sedangkan pada penelitian sebelumnya hanya menggunakan *website*.

2.2. Perbandingan Metode Penyelesaian Masalah

Metode penyelesaian masalah atau lebih dikenal metode penelitian secara umum merupakan suatu metode atau cara yang spesifik yang akan digunakan untuk memecahkan suatu masalah. Pengertian metodologi penelitian menurut adalah pembahasan mengenai konsep teoritik berbagai metode, kelebihan dan kekurangan, yang dalam karya ilmiah dilanjutkan dengan pemilihan metode yang digunakan. Pengertian metodologi adalah “pengkajian terhadap langkah-langkah dalam menggunakan sebuah metode”.

Ada beberapa metode perbandingan yang ada dalam pengerjaan sistem ini yang terdiri dari:

Tabel 2. 1 Perbandingan Metode Penyelesaian Masalah

No	Metode	Pengertian	Kelebihan	Kekurangan
1	Metode <i>Prototype</i> (Roger S, 2002)	Metode <i>Prototype</i> merupakan suatu paradigma baru dalam metode pengembangan perangkat lunak dimana metode ini tidak hanya sekedar evolusi dalam dunia pengembangan	a. Pelanggan berpartisipasi aktif dalam pengembangan sistem, sehingga hasil produk pengembangan akan semakin mudah disesuaikan	1. Proses analisis dan perancangan terlalu singkat. 2. Biasanya kurang fleksibel dalam menghadapi perubahan. 3. Walaupun pemakai melihat berbagai perbaikan dari setiap versi <i>prototype</i> , tetapi pemakai

No	Metode	Pengertian	Kelebihan	Kekurangan
		<p>perangkat lunak, tetapi juga merevolusi metode pengembangan perangkat lunak yang lama yaitu sistem sekuensial yang biasa dikenal dengan nama SDLC atau <i>waterfall development model</i>.</p>	<p>dengan keinginan dan kebutuhan pelanggan.</p> <p>b. Penentuan kebutuhan lebih mudah diwujudkan.</p> <p>c. Mempersingkat waktu pengembangan produk perangkat lunak.</p> <p>d. Adanya komunikasi yang baik antara pengembang dan pelanggan.</p> <p>e. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.</p> <p>f. Lebih menghemat waktu dalam pengembangan sistem.</p> <p>g. Penerapannya menjadi lebih mudah karena pelanggan mengetahui apa yang diharapkannya.</p>	<p>mungkin tidak menyadari bahwa versi tersebut dibuat tanpa memperhatikan kualitas dan pemeliharaan jangka panjang.</p> <p>4. Pengembang kadang-kadang membuat kompromi implementasi dengan menggunakan sistem operasi yang tidak relevan dan algoritma yang tidak efisien.</p>
2	<i>Model</i>	<i>Rapid</i>		

No	Metode	Pengertian	Kelebihan	Kekurangan
	<p><i>Rapid Application Development</i> (RAD) (Kendall, 2010)</p>	<p><i>Application Development</i> (RAD) adalah sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang pendek (kira-kira 60 sampai 90 hari). Model RAD ini merupakan sebuah adaptasi “kecepatan tinggi” dari model sekuensial linier dimana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen.</p>	<ol style="list-style-type: none"> 1. Lebih efektif dari Pengembangan Model <i>waterfall/sequential linear</i> dalam menghasilkan sistem yang memenuhi kebutuhan langsung dari pelanggan. 2. Cocok untuk proyek yang memerlukan waktu yang singkat. 3. Model RAD mengikuti tahap pengembangan sistem seperti pada umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada sehingga pengembang tidak perlu membuatnya dari awal lagi sehingga waktu pengembangan menjadi lebih singkat 	<ol style="list-style-type: none"> 1. Model RAD menuntut pengembangan dan pelanggan memiliki komitmen di dalam aktivitas rapid-fire yang diperlukan untuk melengkapi sebuah sistem, di dalam kerangka waktu yang diperpendek. Jika komitmen tersebut tidak ada, proyek RAD akan gagal. 2. Tidak semua aplikasi sesuai untuk RAD, bila system tidak dapat dimodulkan dengan teratur, pembangunan komponen penting pada RAD akan menjadi bermasalah. 3. RAD tidak cocok digunakan untuk sistem yang mempunyai resiko teknik yang tinggi. 4. Membutuhkan Tenaga kerja yang banyak untuk menyelesaikan sebuah proyek dalam skala besar.

No	Metode	Pengertian	Kelebihan	Kekurangan
			dan efisien.	
3	<i>Waterfall Model</i> (Roger S, 2002)	Model Sekuensial Linier atau sering disebut Model Pengembangan Air Terjun, merupakan paradigma model pengembangan perangkat lunak paling tua, dan paling banyak dipakai. Model ini mengusulkan sebuah pendekatan perkembangan perangkat lunak yang sistematis dan sekuensial yang dimulai pada tingkat dan kemajuan sistem pada seluruh tahapan analisis, desain, kode, pengujian, dan pemeliharaan.	<ol style="list-style-type: none"> 1. Tahapan proses pengembangan lainnya tetap (pasti), mudah diaplikasikan, dan prosesnya teratur. 2. Cocok digunakan untuk produk <i>software/program</i> yang sudah jelas kebutuhannya di awal, sehingga minim kesalahannya. 3. <i>Software</i> yang dikembangkan dengan metode ini biasanya menghasilkan kualitas yang baik. 4. Dokumen pengembangan sistem terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah 	<ol style="list-style-type: none"> 1. Proyek yang sebenarnya jarang mengikuti alur sekuensial seperti diusulkan, sehingga perubahan yang terjadi dapat menyebabkan hasil yang sudah didapatkan tim pengembang harus diubah kembali sering menyebabkan masalah baru. 2. Terjadinya pembagian proyek menjadi tahap-tahap yang tidak fleksibel, karena komitmen harus dilakukan pada tahap awal proses. 3. Sulit untuk mengalami perubahan kebutuhan yang diinginkan oleh <i>customer/pelanggan</i>. 4. Pelanggan harus sabar untuk menanti produk selesai, karena dikerjakan tahap per tahap, dan proses pengerjaannya akan berlanjut ke setiap tahapan bila tahap sebelumnya sudah benar-benar selesai. 5. Perubahan ditengah-tengah pengerjaan produk akan

No	Metode	Pengertian	Kelebihan	Kekurangan
			ke fase berikutnya.	<p>membuat bingung tim pengembang yang sedang membuat produk.</p> <p>6. Adanya waktu kosong (menganggur) bagi pengembang, karena harus menunggu anggota tim proyek lainnya menuntaskan pekerjaannya.</p>
4	Metode SDLC (Roger S, 2002)	siklus untuk membangun sistem dan memberikannya kepada pengguna melalui tahapan perencanaan, analisa, perancangan dan implementasi dengan cara memahami dan menyeleksi keadaan serta proses yang dilakukan pengguna untuk dapat mendukung kebutuhan penggunaanya.	<p>1. Mudah diaplikasikan.</p> <p>2. Memberikan <i>template</i> tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan .</p>	<p>1. Jarang sekali proyek riil mengikuti aliran sekuensial yang dianjurkan model karena model ini bisa melakukan itersi tidak langsung.</p> <p>2. Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk mengakomodasi ketidakpastian pada saat awal proyek.</p> <p>3. Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyek dilalui. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.</p> <p>4. Pengembang sering melakukan</p>

N o	Metode	Pengertian	Kelebihan	Kekurangan
				penundaan yang tidak perlu karena anggota tim proyek harus menunggu tim lain untuk melengkapi tugas karena memiliki ketergantungan hal ini menyebabkan penggunaan waktu tidak efisien.

2.3. Landasan Teori

2.2.1. AIML

Artificial Intelligence Markup Language (AIML) adalah sebuah bahasa yang mendeskripsikan objek data dan perilaku program komputer yang memprosesnya. AIML sendiri merupakan turunan dari *Extensible Markup Language* (XML). Objek AIML tersusun atas unit-unit yang disebut *topics* dari *categories*, berisi data yang sudah ter-*parsing* maupun belum ter-*parsing*. Data yang ter-*parsing* berisi karakter-karakter, beberapa di antaranya berupa data karakter, yang lainnya dapat berupa elemen AIML. Elemen AIML menangkap pengetahuan dalam bentuk stimulus-response di dokumen (Suryani & Amalia, 2017).

AIML berisi kumpulan pola dan respon yang dapat digunakan oleh *chatbot* untuk penelusuran jawaban setiap kalimat yang diberikan. *Interpreter* AIML diperlukan untuk menerima *input* dan melakukan penelusuran jawaban pada dokumen AIML. Saat ini tersedia banyak interpreter AIML dalam berbagai

bahasa pemrograman sehingga proses pembuatan *chatbot* dapat terfokus pada penyusunan dokumen AIML. Selain itu juga saat ini di internet banyak tersedia dokumen AIML siap pakai untuk berbagai bidang percakapan. Bagian-bagian penting dari AIML terdiri dari : AIML, *Pattern*, *template* dan *That*. (Suryani & Amalia, 2017)

2.2.2. Chatbot

Chatbot adalah program komputer yang berinteraksi dengan pengguna menggunakan bahasa alami. Teknologi ini dimulai pada tahun 1960, tujuannya adalah untuk melihat apakah sistem *chatbot* dapat mengelabui pengguna bahwa mereka adalah manusia sejati. Namun, sistem *chatbot* tidak hanya dibangun untuk meniru percakapan manusia, dan menghibur pengguna. *Chatbot* juga dapat digunakan dalam bidang pendidikan, retrieval informasi, bisnis, dan e-commerce. (Suryani & Amalia, 2017)

2.2.3. Artificial Intelligence (AI)

Pada dasarnya AI adalah suatu pengetahuan yang membuat komputer dapat meniru kecerdasan manusia sehingga komputer dapat melakukan hal-hal yang dikerjakan manusia dimana membutuhkan suatu kecerdasan, misalkan melakukan analisis penalaran untuk mengambil suatu kesimpulan atau keputusan atau penerjemahan dari satu bahasa ke bahasa lain (Herman Firmansyah, 2009).

Algoritma pencarian yang digunakan pencarian dan pelacakan merupakan salah satu hal penting yang menentukan keberhasilan dan kesuksesan sebuah kecerdasan buatan. Pada dasarnya terdapat 2 teknik pencarian dan pelacakan yang

digunakan dalam kecerdasan buatan, yaitu Pencarian Buta (*Blind Search*) dan Pencarian Heuristik (*Heuristic Search*). (Herman Firmansyah, 2009)

2.2.4. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*Database Management System*), database ini *multithread* dan *multi user*. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi *GNU General Public License* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus yang bersifat khusus. (Bunafit, 2010).

Kekuatan MySQL tidak ditopang oleh sebuah komunitas, seperti *Apache*, yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh pemilik masing-masing, tetapi MySQL didukung penuh oleh sebuah perusahaan profesional dan komersial, yakni MySQL AB dari Swedia (Wardani, 2012).

MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (General Public License). Di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed Source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*) (Wardani, 2012).

2.2.5.OOAD (*Object Oriented Analysis & Design*)

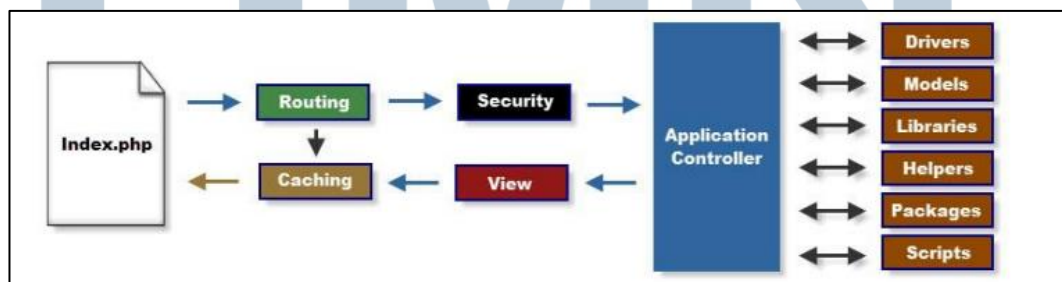
Object-Oriented Analysis and Design merupakan analisis dan perancangan berorientasi objek, suatu kumpulan petunjuk umum untuk membuat analisis dan desain. (Lars Mathiassen, 2000)

2.2.6.PHP

PHP adalah bahas pemrograman untuk membuat halaman web yang dinamis. Cara kerja *PHP* adalah dengan menyelipkan di antara kode *HTML* (*hypertext markup language*). *PHP* bersifat terbuka dan *multiplatform*, karenanya dapat dijalankan di banyak merek web server. Saat ini, pengguna *PHP* sudah banyak, diklaim ada lebih dari dua puluh juta *website* yang menggunakan bahasa pemrograman *PHP* dijalankan lebih dari satu juta server. (Ali Zaki, 2008)

2.2.7.Framework CodeIgniter

CodeIgniter adalah aplikasi *open source* berupa *framework* dengan model MVC (Supono, 2018). *CodeIgniter* atau *CI* merupakan framework untuk pengembangan aplikasi berbasis web dengan menggunakan script *PHP*.



Gambar 2. 1 Prinsip Kerja CodeIgniter

Sumber : (Supono, 2018)

Prinsip kerja CI adalah dimulai pada *file* index.php. *file* ini sebagai pengendali utama yang berfungsi memuat kode *script* utama yang berfungsi menjalankan CodeIgniter. Pada saat client mengakses *file* index.php maka akan diteruskan oleh proses *routing*. Pada bagian *routing* ini akan dicek terlebih dahulu apakah *file* yang diminta ada di bagian *caching* atau tidak, jika ada maka akan diambilkan dari *caching*, namun jika tidak ada maka akan dilanjutkan masuk ke dalam pengecekan kemandirian. Proses selanjutnya adalah mengakses pada *controller* dari *framework* CI. Pada bagian ini akan dihubungkan ke komponen model, *driver* *database*, *library* dan lain sebagainya yang kemudian akan diarahkan pada *file* *view* yang diminta oleh *client* (Dudi Rahmandiansyah, 2012).

2.2.8. Use Case

Use case merupakan pemodelan untuk (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada Di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa, 2013).

2.2.9. Activity diagram

Activity diagram digunakan untuk menggambarkan alur aktivitas / kegiatan dari proses bisnis, juga untuk memodelkan respon yang akan terjadi apabila suatu operasi dalam sistem dijalankan (P. Clements, 2011).

2.2.10. Class Diagram

Class Diagram menggambarkan kumpulan dari *class-class* yang ada serta hubungan diantara *class* tersebut. *Class* mempunyai *attributes* dan *operations* yang berbeda-beda (Lars Mathiassen, 2000).

